

Emacs 快捷键列表

C = Control

M = Meta = Alt | Esc

Del = Backspace

基本快捷键(Basic)

C-x C-f "find"文件, 即在缓冲区打开/新建一个文件

C-x C-s 保存文件

C-x C-w 使用其他文件名另存为文件

C-x C-v 关闭当前缓冲区文件并打开新文件

C-x i 在当前光标处插入文件

C-x b 新建/切换缓冲区

C-x C-b 显示缓冲区列表

C-x k 关闭当前缓冲区

C-z 挂起 emacs

C-x C-c 关闭 emacs

光标移动基本快捷键(Basic Movement)

C-f 后一个字符

C-b 前一个字符

C-p 上一行

C-n 下一行

M-f 后一个单词

M-b 前一个单词

C-a 行首

C-e 行尾

C-v 向下翻一页

M-v 向上翻一页

M-< 到文件开头

M-> 到文件末尾

编辑(Editint)

M-n 重复执行后一个命令 n 次

C-u 重复执行后一个命令 4 次

C-u n 重复执行后一个命令 n 次

C-d 删除(delete)后一个字符

M-d 删除后一个单词

Del 删除前一个字符

M-Del 删除前一个单词

C-k 移除(kill)一行

C-Space 设置开始标记 (例如标记区域)
C-@ 功能同上, 用于 C-Space 被操作系统拦截的情况
C-w 移除(kill)标记区域的内容
M-w 复制标记区域的内容
C-y 召回(yank)复制/移除的区域/行
M-y 召回更早的内容 (在 kill 缓冲区内循环)
C-x C-x 交换光标和标记

C-t 交换两个字符的位置
M-t 交换两个单词的位置
C-x C-t 交换两行的位置
M-u 使从光标位置到单词结尾处的字母变成大写
M-l 与 M-u 相反
M-c 使从光标位置开始的单词的首字母变为大写

重要快捷键(Important)

C-g 停止当前运行/输入的命令
C-x u 撤销前一个命令
M-x revert-buffer RETURN (照着这个输入)撤销上次存盘后所有改动
M-x recover-file RETURN 从自动存盘文件恢复
M-x recover-session RETURN 如果你编辑了几个文件, 用这个恢复

在线帮助(Online-Help)

C-h c 显示快捷键绑定的命令
C-h k 显示快捷键绑定的命令和它的作用
C-h l 显示最后 100 个键入的内容
C-h w 显示命令被绑定到哪些快捷键上
C-h f 显示函数的功能
C-h v 显示变量的含义和值
C-h b 显示当前缓冲区所有可用的快捷键
C-h t 打开 emacs 教程
C-h i 打开 info 阅读器
C-h C-f 显示 emacs FAQ
C-h p 显示本机 Elisp 包的信息

搜索/替换(Search/Replace)

C-s 向后搜索
C-r 向前搜索
C-g 回到搜索开始前的位置(如果你仍然在搜索模式中)
M-% 询问并替换(query replace)

Space 或 y 替换当前匹配
Del 或 n 不要替换当前匹配
. 仅仅替换当前匹配并退出(替换)
, 替换并暂停(按 Space 或 y 继续)

! 替换以下所有匹配
^ 回到上一个匹配位置
RETURN 或 q 退出替换

使用正则表达式(Regular expression)搜索/替换

可在正则表达式中使用的符号:

^ 行首
\$ 行尾
. 单个字符
.* 任意多个(包括没有)字符
\< 单词开头
\> 单词结尾
[] 括号中的任意一个字符(例如[a-z]表示所有的小写字母)

M C-s RETURN 使用正则表达式向后搜索

M C-r RETURN 使用正则表达式向前搜索

C-s 增量搜索

C-s 重复增量搜索

C-r 向前增量搜索

C-r 重复向前增量搜索

M-x query-replace-regexp 使用正则表达式搜索并替换

窗口命令(Window Commands)

C-x 2 水平分割窗格

C-x 3 垂直分割窗格

C-x o 切换至其他窗格

C-x 0 关闭窗格

C-x 1 关闭除了光标所在窗格外所有窗格

C-x ^ 扩大窗格

M-x shrink-window 缩小窗格

M C-v 滚动其他窗格内容

C-x 4 f 在其他窗格中打开文件

C-x 4 0 关闭当前缓冲区和窗格

C-x 5 2 新建窗口(frame)

C-x 5 f 在新窗口中打开文件

C-x 5 o 切换至其他窗口

C-x 5 0 关闭当前窗口

书签命令(Bookmark commands)

C-x r m 在光标当前位置创建书签

C-x r b 转到书签

M-x bookmark-rename 重命名书签

M-x bookmark-delete 删除书签

M-x bookmark-save 保存书签

C-x r l 列出书签清单

- d 标记等待删除
- Del 取消删除标记
- x 删除被标记的书签
- r 重命名
- s 保存列表内所有书签
- f 转到当前书签指向的位置
- m 标记在多窗口中打开
- v 显示被标记的书签(或者光标当前位置的书签)
- t 切换是否显示路径列表
- w 显示当前文件路径
- q 退出书签列表

M-x bookmark-write 将所有书签导出至指定文件

M-x bookmark-load 从指定文件导入书签

Shell

M-x shell 打开 shell 模式

C-c C-c 类似 unix 里的 C-c(停止正在运行的程序)

C-d 删除光标后一个字符

C-c C-d 发送 EOF

C-c C-z 挂起程序(unix 下的 C-z)

M-p 显示前一条命令

M-n 显示后一条命令

DIrectory EEditor (dired)

C-x d 打开 dired

C(大写 C) 复制

d 标记等待删除

D 立即删除

e 或 f 打开文件或目录

g 刷新当前目录

G 改变文件所属组(chgrp)

k 从屏幕上的列表里删除一行(不是真的删除)

m 用*标记

n 光标移动到下一行

o 在另一个窗格打开文件并移动光标

C-o 在另一个窗格打开文件但不移动光标

P 打印文件

q 退出 dired

Q 在标记的文件中替换

R 重命名文件

u 移除标记

v 显示文件内容

x 删除有 D 标记的文件

Z 压缩/解压缩文件

M-De1 移除标记(默认为所有类型的标记)
~ 标记备份文件(文件名有~的文件)等待删除
标记自动保存文件(文件名形如#name#)等待删除
*/ 用*标记所有文件夹(用 C-u */n 移除标记)
= 将当前文件和标记文件(使用 C-@标记而不是 dired 的 m 标记)比较
M-= 将当前文件和它的备份比较
! 对当前文件应用 shell 命令
M-} 移动光标至下一个用*或 D 标记的文件
M-{ 移动光标至上一个用*或 D 标记的文件
% d 使用正则表达式标记文件等待删除
% m 使用正则表达式标记文件为*
+ 新建文件夹
> 移动光标至后一个文件夹
< 移动光标至前一个文件夹
s 切换排序模式(按文件名/日期)

或许把这个命令归入这一类也很合适:

M-x speedbar 打开一个独立的目录显示窗口

Telnet

M-x telnet 打开 telnet 模式
C-d 删除后一个字符或发送 EOF
C-c C-c 停止正在运行的程序(和 unix 下的 C-c 类似)
C-c C-d 发送 EOF
C-c C-o 清除最后一个命令的输出
C-c C-z 挂起正在运行的命令
C-c C-u 移除前一行
M-p 显示前一条命令

Text

只能在 text 模式里使用

M-s 使当前行居中
M-S 使当前段落居中
M-x center-region 使被选中的区域居中

宏命令(Macro-commands)

C-x (开始定义宏
C-x) 结束定义宏
C-x e 运行最近定义的宏
M-n C-x e 运行最近定义的宏 n 次
M-x name-last-kbd-macro 给最近定义的宏命名(用来保存)
M-x insert-kbd-macro 将已命名的宏保存到文件
M-x load-file 载入宏

编程(Programming)

M C-\ 自动缩进光标和标记间的区域
M-m 移动光标到行首第一个(非空格)字符
M-^ 将当前行接到上一行末尾处
M-; 添加缩进并格式化的注释
C, C++和 Java 模式
M-a 移动光标到声明的开始处
M-e 移动光标到声明的结尾处
M C-a 移动光标到函数的开始处
M C-e 移动光标到函数的结尾处
C-c RETURN 将光标移动到函数的开始处并标记到结尾处
C-c C-q 根据缩进风格缩进整个函数
C-c C-a 切换自动换行功能
C-c C-d 一次性删除光标后的一串空格(greedy delete)

为了实现下面的一些技术, 你需要在保存源代码的目录里运行“etags
*.c *.h *.cpp”(或者源代码的其他的扩展名)

M-. (点) 搜索标签

M-x tags-search ENTER 在所有标签里搜索(使用正则表达式)

M-, (逗号) 在 tags-search 里跳至下一个匹配处

M-x tags-query-replace 在设置过标签的所有文件里替换文本

GDB(调试器)

M-x gdb 在另一个的窗格中打开 gdb

版本控制(Version Control)

C-x v d 显示当前目录下所有注册过的文件(show all registered files in this dir)

C-x v = 比较不同版本间的差异(show diff between versions)

C-x v u 移除上次提交之后的更改(remove all changes since last checkin)

C-x v ~ 在不同窗格中显示某个版本(show certain version in different window)

C-x v l 打印日志(print log)

C-x v i 标记文件等待添加版本控制(mark file for version control add)

C-x v h 给文件添加版本控制文件头(insert version control header into file)

C-x v r 获取命名过的快照(check out named snapshot)

C-x v s 创建命名的快照(create named snapshot)

C-x v a 创建 gnu 风格的更改日志(create changelog file in gnu-style)

Emacs Commands List

C = Control

M = Meta = Alt|Esc

Basics

C-x C-f "find" file i.e. open/create a file in buffer

C-x C-s save the file
C-x C-w write the text to an alternate name
C-x C-v find alternate file
C-x i insert file at cursor position
C-x b create/switch buffers
C-x C-b show buffer list
C-x k kill buffer
C-z suspend emacs
C-X C-c close down emacs

Basic movement

C-f forward char
C-b backward char
C-p previous line
C-n next line
M-f forward one word
M-b backward one word
C-a beginning of line
C-e end of line
C-v one page up
M-v scroll down one page
M-< beginning of text
M-> end of text

Editing

M-n repeat the following command n times
C-u repeat the following command 4 times
C-u n repeat n times
C-d delete a char
M-d delete word
M-Delete delete word backwards
C-k kill line

C-Space Set beginning mark (for region marking for example)
C-W "kill" (delete) the marked region
M-W copy the marked region
C-y "yank" (paste) the copied/killed region/line
M-y yank earlier text (cycle through kill buffer)
C-x C-x exchange cursor and mark

C-t transpose two chars
M-t transpose two words
C-x C-t transpose lines
M-u make letters uppercase in word from cursor position to end
M-c simply make first letter in word uppercase
M-l opposite to M-u

Important

C-g quit the running/entered command
C-x u undo previous action
M-x revert-buffer RETURN (insert like this) undo all changes since last save
M-x recover-file RETURN Recover text from an autosave-file
M-x recover-session RETURN if you edited several files

Online-Help

C-h c which command does this keystroke invoke
C-h k which command does this keystroke invoke and what does it do?
C-h l what were my last 100 typed keys
C-h w what key-combo does this command have?
C-h f what does this function do
C-h v what's this variable and what is its value
C-h b show all key commands for this buffer
C-h t start the emacs tutorial
C-h i start the info reader
C-h C-k start up info reader and go to a certain key-combo point
C-h F show the emacs FAQ
C-h p show info about the Emacs package on this machine

Search/Replace

C-s Search forward
C-r search backward
C-g return to where search started (if you are still in search mode)
M-% query replace

Space or y replace this occurrence
Del or n don't replace
. only replace this and exit (replace)
, replace and pause (resume with Space or y)
! replace all following occurrences
^ back to previous match
RETURN or q quit replace

Search/Replace with regular expressions

Characters to use in regular expressions:

^ beginning of line
\$ end of line
. single char
.* group or null of chars
\< beginning of a word
\> end of a word
[] every char inside the brackets (for example [a-z] means every small letter)

M C-s RETURN search for regular expression forward
M C-r RETURN search for regular expression backward

M C-s incremental search
C-s repeat incremental search
M C-r incremental search backwards
C-r repeat backwards
M-x query-replace-regexp search and replace

Window-Commands

C-x 2 split window vertically
C-x o change to other window
C-x 0 delete window
C-x 1 close all windows except the one the cursors in
C-x ^ enlarge window
M-x shrink-window command says it ;-)
M C-v scroll other window
C-x 4 f find file in other window
C-x 4 o change to other window
C-x 4 0 kill buffer and window
C-x 5 2 make new frame
C-x 5 f find file in other frame
C-x 5 o change to other frame
C-x 5 0 close this frame

Bookmark commands

C-x r m set a bookmark at current cursor pos
C-x r b jump to bookmark
M-x bookmark-rename says it
M-x bookmark-delete "
M-x bookmark-save"
C-x r l list bookmarks

d mark bookmark for deletion
r rename bookmark
s save all listed bookmarks
f show bookmark the cursor is over
m mark bookmarks to be shown in multiple window
v show marked bookmarks (or the one the cursor is over)
t toggle listing of the corresponding paths
w " path to this file
x delete marked bookmarks
Del ?
q quit bookmark list

M-x bookmark-write write all bookmarks in given file
M-x bookmark-load load bookmark from given file

Shell

M-x shell starts shell modus
C-c C-c same as C-c under unix (stop running job)
C-d delete char forward
C-c C-d Send EOF
C-c C-z suspend job (C-z under unix)
M-p show previous commands

DIRectory EDitor (dired)

C-x d start up dired
C (large C) copy
d mark for erase
D delete right away
e or f open file or directory
g reread directory structure from file
G change group permissions (chgrp)
k delete line from listing on screen (don't actually delete)
m mark with *
n move to next line
o open file in other window and go there
C-o open file in other window but don't change there
P print file
q quit dired
Q do query-replace in marked files
R rename file
u remove mark
v view file content
x delete files marked with D
z compress file
M-De l remove all marks (whatever kind)
~ mark backup files (name~ files) for deletion
mark auto-save files (#name#) for deletion
*/ mark directory with * (C-u * removes that mark again)
= compare this file with marked file
M-= compare this file with it's backup file
! apply shell command to this file
M-} change to the next file marked with * od D
M-{ "previous"
% d mark files described through regular expression for deletion
% m " (with *)
+ create directory
> changed to next dir
< change to previous dir
s toggle between sorting by name or date

Maybe into this category also fits this command:

M-x speedbar starts up a separate window with a directory view

Telnet

M-x telnet starts up telnet-modus
C-d either delete char or send EOF
C-c C-c stop running job (similar to C-c under unix)
C-c C-d send EOF
C-c C-o clear output of last command
C-c C-z suspend execution of command
C-c C-u kill line backwards
M-p recall previous command

Text

Works only in text mode

M-s center line
M-S center paragraph
M-x center-region name says

Macro-commands

C-x (start macro definition
C-x) end of macro definition
C-x e execute last defined macro
M-n C-x e execute last defined macro n times
M-x name-last-kbd-macro give name to macro (for saving)
M-x insert-keyboard-macro save named macro into file
M-x load-file load macro
M-x macroname execute macroname

Programming

M C-\ indent region between cursor and mark
M-m move to first (non-space) char in this line
M-^ attach this line to previous
M-; formatize and indent comment
C, C++ and Java Modes
M-a beginning of statement
M-e end of statement
M C-a beginning of function
M C-e end of function
C-c RETURN Set cursor to beginning of function and mark at the end
C-c C-q indent the whole function according to indentation style
C-c C-a toggle modus in which after electric signs (like {}:'./*) emacs does the indentation
C-c C-d toggle auto hungry mode in which emacs deletes groups of spaces with one del-press
C-c C-u go to beginning of this preprocessor statement
C-c C-c comment out marked area

More general (I guess)

M-x outline-minor-mode collapses function definitions in a file to a mere {...}
M-x show-subtree If you are in one of the collapsed functions, this un-collapses it

In order to achieve some of the features coming up now you have to run `etags *.c *.h *.cpp` (or whatever ending your source files have) in the source directory

`M-.` (That's Meta dot) If you are in a function call, this will take you to its definition

`M-x tags-search` `ENTER` Searches through all your etaged

`M-,` (Meta comma) jumps to the next occurrence for `tags-search`

`M-x tags-query-replace` `yum`. This lets you replace some text in all the tagged files

GDB (Debugger)

`M-x gdb` starts up `gdbm` in an extra window

Version Control

`C-x v d` show all registered files in this dir

`C-x v =` show diff between versions

`C-x v u` remove all changes since last checkin

`C-x v ~` show certain version in different window

`C-x v l` print log

`C-x v i` mark file for version control add

`C-x v h` insert version control header into file

`C-x v r` check out named snapshot

`C-x v s` create named snapshot

`C-x v a` create changelog file in gnu-style